



NRL/MR/5520--94-7601

Discrete-Event-Dynamic-System-Based Approaches for Control in Integrated Voice/Data Multihop Radio Networks

CHRISTOS C. CASSANDRAS
VIBHOR JULKA

*University of Massachusetts
Amherst, MA*

Prepared for

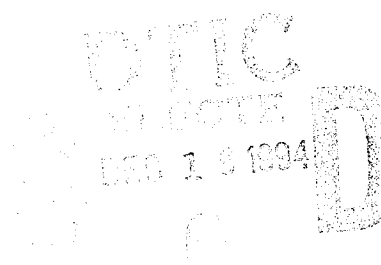
*Communication Systems Branch
Information Technology Division*

December 7, 1994

19941213 008

Approved for public release; distribution unlimited.

DTIC QUALITY INSPECTED 1



REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE December 7, 1994	3. REPORT TYPE AND DATES COVERED Final Report 10/92-9/93		
4. TITLE AND SUBTITLE Discrete-Event-Dynamic-System-Based Approaches for Control in Integrated Voice/Data Multihop Radio Networks			5. FUNDING NUMBERS PE - 61153N PR - RR015-09-41 WU - DN159-036	
6. AUTHOR(S) Christos G. Cassandras* and Vibhor Julka*				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Research Laboratory Washington, DC 20375-5320			8. PERFORMING ORGANIZATION REPORT NUMBER NRL/MR/5520-94-7601	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research 800 North Quincy Street Arlington, VA 22217-5660			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES *Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA Prepared for Communication Systems Branch, Information Technology Division				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) We report accomplishments and new directions in our effort to develop and apply discrete-event-dynamic-system-based techniques for the transmission scheduling problem in Radio Networks (RN). First, we examine this problem in the context of data traffic in general topology networks. Next, we look at the scheduling problem when processing packetized voice calls, where Grade-of-Service (GOS) requirements are quite different. For data traffic, we formulate an optimization problem for the allocation of transmission time slots to different competing nodes and present a gradient-based algorithm suitable for on-line implementation without any assumptions on the nature of the data traffic processes. Examples illustrating the adaptive features of our approach and comparing it to other schemes are included. For voice traffic, we formulate the optimal scheduling problem as a stochastic discrete resource allocation problem, which is combinatorially hard. We describe a technique for transforming this to a continuous optimization problem and develop algorithms for minimizing the blocking probability of packetized voice traffic.				
14. SUBJECT TERMS Communications network Voice/data integration Discrete-event dynamic systems Optimization Link activation Perturbation analysis			15. NUMBER OF PAGES 34	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

CONTENTS

1.	INTRODUCTION	1
2.	TRANSMISSION SCHEDULING PROBLEM FORMULATION WITH OVERLAPPING TRANSMISSION SETS	4
3.	A SCHEDULE OPTIMIZATION ALGORITHM	9
4.	A DETERMINISTIC SCHEDULING POLICY	12
5.	SCHEDULING IN NETWORKS WITH VOICE TRAFFIC	15
6.	DISCRETE OPTIMIZATION PROBLEM FORMULATION	17
7.	DERIVATIVE ESTIMATION THROUGH THE MARKED SLOT APPROACH	19
8.	CONVERGENCE TO A GLOBAL OPTIMUM	22
9.	OPTIMAL TDM FRAME DESIGN	23
10.	OPTIMAL ADMISSION CONTROL IN BLOCKING NETWORKS	23
11.	CONCLUSIONS AND FUTURE WORK	27
	REFERENCES	30

Accession For	
NTIS CRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

DISCRETE-EVENT-DYNAMIC-SYSTEM-BASED APPROACHES FOR CONTROL IN INTEGRATED VOICE/DATA MULTIHOP RADIO NETWORKS

1 Introduction

This report contains the results from our work under contract N000014-92-J-2017 for the period 10/1/92 - 9/30/93. It describes the progress made towards the effort to develop and apply new discrete-event-dynamic-system-based techniques for the transmission scheduling problem in Radio Networks (RN). The report is in two parts: in part one we examine the transmission scheduling problem in the context of (fixed length) data traffic. In contrast to our earlier work, performed under contract N000014-91-J-2025 [3, 4], which dealt with the broadcast scheduling problem in a fully connected packet RN, the first part of this report extends the proposed methodology therein to general topology networks. In part two, we look at the scheduling problem when processing packetized voice calls¹ in a N -node multihop RN. As will be seen, the different *Grade-of-Service (GOS)* requirements associated with voice and data traffic lead to two decidedly different scheduling policies. We begin with a brief review of the transmission scheduling problem in N -node Radio Networks.

In such networks there are N nodes that wish to broadcast over a common channel. In order for a node to successfully transmit, the following *primary interference constraint* must be satisfied: *if node i transmits, then all nodes that are within node i 's broadcast zone must be in a receiving mode*. Thus, all such nodes j are viewed as "neighbors" of i , connected to i through links (i, j) . In addition, there may be a *secondary interference constraint*, whereby if i is transmitting and j is connected to i , then all nodes k that are also connected to j must be in a receiving mode; otherwise, a (k, j) transmission interferes with the (i, j) transmission. In short, there are (generally overlapping) subsets of the N nodes, known as *transmission sets*, such that all nodes in such a set can simultaneously transmit without conflict (i.e. without violating the interference constraints). Given a slotted time line, the problem becomes the assignment of each time slot to a particular transmission set [10, 9, 18, 20, 23] so that some desirable performance can be attained.

There have been several approaches for solving the transmission scheduling problem (see references in [3]), however here, as in [3, 4], an alternate approach is proposed which uses *on-line adaptation* of the schedule (in a stochastic operating environment), to achieve the desired system performance. An important distinction between the existing literature and the work reported here and in [4] is that *no distributional information at all about the arrival processes at the various network nodes is required*.

As described in [4], the transmission scheduling problem can be placed in the context of a "polling" system where the channel is modelled as a single resource (server) which provides service to a set C of transmission sets. At each network node i , $i = 1, \dots, N$, there is a customer² arrival stream characterized by arbitrary interarrival time distributions G_i with rate λ_i . Let A denote the set of arrival streams. Note that we allow for correlation within an individual queue arrival stream,

¹In this report it is assumed a voice call is comprised of an arbitrary number of fixed-length packets, with constant interarrival times.

²The term "customer" can refer either to a data packet or to an entire voice call.

as well as between streams. A customer from a particular arrival stream is allowed to join one of several prespecified transmission sets associated with that stream. Therefore, a transmission set C_i is such that $C_i \subseteq A$ and, in general, $C_i \cap C_j \neq \emptyset$ for $i \neq j$. The importance of the distinction

between "transmission set" and "arrival streams" is the following: when the server is allocated to a transmission set, it can serve multiple customers simultaneously, one from every arrival stream that can belong to this set. As an illustration, consider the four-node RN with three transmission sets shown in Figure 1a. The equivalent polling representation is shown in Figure 1b, where, a customer from arrival stream 2, for example, may join either transmission set 1 or 2. In the latter model, a customer is held at the head of its arrival stream queue until the server is assigned to either transmission set 1 or 2; it is then instantaneously transferred to the appropriate queue and served (while another customer from another arrival stream may also receive service at the same time). We assume that time is slotted (with slot size set equal to the packet transmission time), such that a scheduling decision is made at the beginning of every slot. The rule by which the server is assigned to a particular queue is known as a *polling policy*.

The choice of an appropriate polling policy is governed in part by the traffic GOS requirements. Data traffic is characterized by a tolerance in the end-to-end packet delays, but no tolerance in packet loss. Therefore, in packet RN we assume infinite buffers at each network node and such that packets can be queued with no additional penalties. Our objective is then to select a polling policy so as to minimize the mean packet waiting time. In contrast, packetized voice traffic can tolerate some loss (without a severe deterioration in the voice playback), however it is sensitive to end-to-end delays. In addition, it is desirable that packets reach the destination in the order in which they are transmitted. Assume that the desired GOS for voice calls can be expressed in the form of a stringent *upper bound on the end-to-end delay of packets belonging to a voice call*. Thus we deal with the problem of serving *delay-sensitive traffic* in the following sense: packets cannot be queued for more than n time slots, where n is fixed. Thus, upon arrival, a voice call is either accepted if this constraint can be satisfied, or it is *blocked*. Blocked calls are assumed to be lost from the system, a mode of operation known as 'blocked calls cleared'. The objective then is to determine the polling policy that minimizes the probability that a voice call is blocked.

For RN's with data traffic, as discussed in [3], we consider a non-cyclic polling policy, in particular, a *non-work conserving*³ *probabilistic policy with zero switchover times*, where the current time slot is assigned to transmission set i with probability θ_i . We refer to this policy as '*random polling*'. The θ_i 's then represent control parameters, which can be *gradually tuned* or *optimally selected* so as to achieve a desired system performance. In scheduling delay sensitive voice traffic we consider a *cyclic policy*, where the polling cycle is specified by a *frame* comprised of n slots, such that a slot is uniquely assigned to a particular transmission set. Thus, a frame is characterized by a vector $[c_1, \dots, c_n]$ where $c_i \in \{1, \dots, M\}$ is the transmission set assigned to slot i and $M = |C|$ where $|\cdot|$ is the cardinality of a set. Note that the frame structure allows us to guarantee an upper bound on the *intervisit time* to any transmission set. In the case of delay-sensitive traffic, we can therefore ensure that no transmission set ever waits for more than n time slots.

In contrast to random polling, the scheduling problem under cyclic polling is formulated as a stochastic *discrete optimization* problem over a finite set, and is an instance of the general stochastic

³A policy is work conserving or nonidling if the server is assigned to a transmission set if and only if there are customers present in one of the arrival streams belonging to that transmission set.

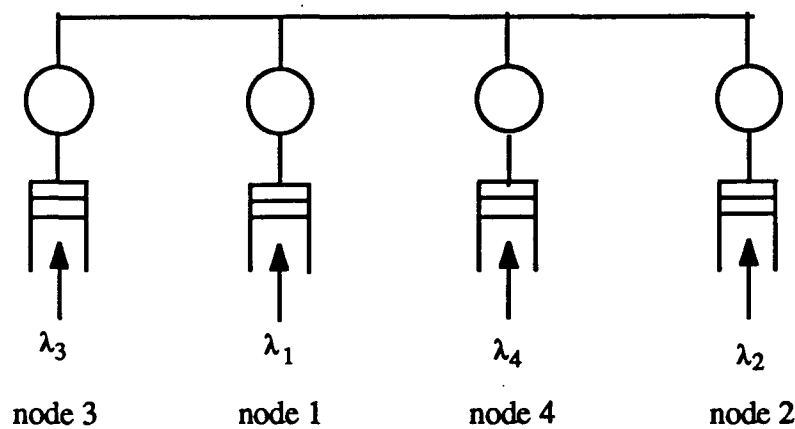


Figure 1a : 4-Node RN

Transmission sets:
 $\{(1,2),(2,3),(3,4)\}$

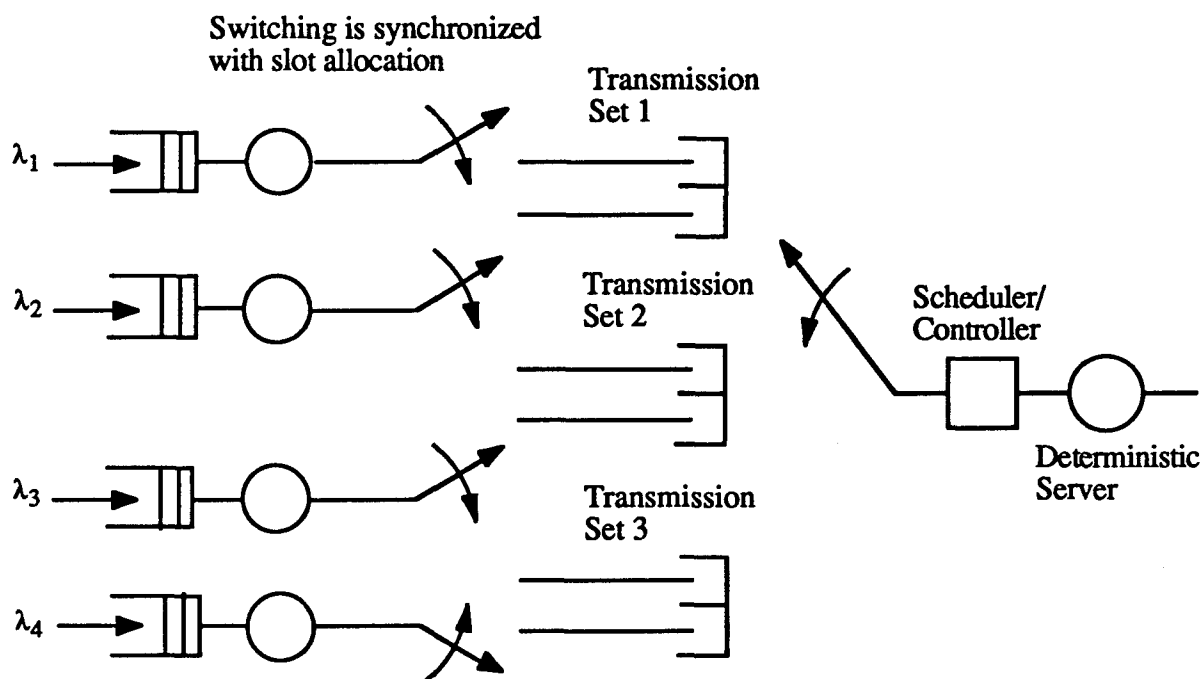


Figure 1b: Multiclass Polling System

resource allocation problem where n resources (*slots*) are to be assigned to M users (*transmission sets*). Rather than solve the original discrete optimization problem, which can easily become combinatorially explosive, we propose a new approach [6] whereby we transform the former into an auxiliary *continuous optimization* problem, which in turn can be solved quite efficiently through various gradient-based algorithms. Finally, the choice of a state-independent scheduling policy is motivated by the simplicity of the controller, i.e., the fact that no monitoring of state information (queue lengths) is involved. In fact, in many complex systems, including radio networks, it is often the case that state information is either impossible to obtain or it may not be up-to-date for a real-time controller to use.

The remainder of the report is organized as follows. In section 2 we present a formulation of the transmission scheduling problem with overlapping transmission sets in the presence of data traffic. This formulation reveals the need for estimates of the sensitivities (gradient) of the performance measure with respect to the slot assignment probability parameters θ_i , based on which simple on-line optimization algorithms are proposed. We present one such estimator for the case where the performance measure is the mean packet waiting time. In section 3, we present a gradient-based optimization algorithm for obtaining optimal slot assignment probabilities using our approach for several networks, including examples illustrating the adaptive features of the approach. In section 4, our aim is to use our approach to subsequently obtain a *deterministic* (non-randomized) scheduling policy, thus avoiding the randomization in the slot assignment process. We develop such a deterministic policy using the *Golden Ratio* policy proposed in [15], and compare it with several other policies, including a simple *Round Robin* one. In section 5 we consider the problem of serving *delay-sensitive voice traffic*. In section 6 we formulate the optimal scheduling problem as a *stochastic resource allocation problem*, and describe how we transform the ensuing discrete optimization problem to a continuous optimization problem. This formulation reveals the need for estimates of the sensitivity (gradient) of performance metrics with respect to the control parameters. In section 7 we derive such estimators based on the ‘marked slot’ approach proposed in [4], and develop explicit algorithms for gradient estimators of blocking probability under two modes of delay-sensitive operation. In section 8 we present a convergence result which delineates the relationship between the discrete optimization problem and the auxiliary continuous optimization problem. In sections 9 and 10, we illustrate our methodology by discussing two applications which can be modelled as resource allocation problems. Lastly in section 11 we discuss ongoing research problems.

2 Transmission Scheduling Problem Formulation with Overlapping Transmission Sets

Consider a N -node RN with data traffic. At each node i , $i = 1, \dots, N$, there is a process characterizing data arrivals, where, as before, we assume an arbitrary interarrival distribution for data packets and allow for correlations between arrivals. From a terminology standpoint, we will use “node” and “arrival stream” interchangeably. Our starting point is to assume that a finite number of transmission sets (M) has been identified such that all nodes in the RN belong to at least one such set. Moreover let there be one or more overlapping transmission sets (i.e., a packet from some arrival stream (node) may be served in transmission set i or $j \neq i$). Let Z_i be the set of transmission sets to which the i th arrival stream belongs. For example, in the 4-node network shown in Figure

1a, node 2 belongs to transmission sets 1 and 2 so that $\mathcal{Z}_2 = \{1, 2\}$. Similarly $\mathcal{Z}_3 = \{2, 3\}$ and so on.

Let us focus on average packet waiting time as the cost function of interest, and let θ_i , $i = 1, \dots, M$ be the probability that a slot is assigned to transmission set i (assignments are made independently from one slot to the next). We define Θ to be the M -dimensional slot assignment probability vector $[\theta_1, \dots, \theta_M]^T$ and $W_i(\Theta)$ as the expected node i waiting time. Our objective is to determine the optimal slot assignment vector so as to minimize a weighted sum of the mean packet waiting times associated with each arrival stream, subject to normalization and stability constraints. Thus, the optimization problem is formulated as:

Problem P1:

$$\min_{\Theta} \sum_{i=1}^N C_i W_i(\Theta) \quad \text{s.t.} \quad \sum_{j=1}^M \theta_j = 1 \quad (1)$$

where C_i is the weight associated with stream i . In the above problem formulation we do not explicitly include the stability constraint, rather, we take this into account in the final optimal assignment by simply ensuring that every adjustment made to θ_i never violates this constraint. Observe that the mean waiting time W_i is a function of the probability that stream i is assigned a transmission slot, which is given by $\sum_{j \in \mathcal{Z}_i} \theta_j$. This motivates the definition of a new N -dimensional control vector Φ with elements

$$\phi_i = \sum_{j \in \mathcal{Z}_i} \theta_j, \quad i = 1, \dots, N \quad (2)$$

that is, ϕ_i is the probability that a time slot is assigned to a transmission set that contains stream i (and hence a packet from stream i , if present, is served). With the introduction of Φ , we can rewrite P1 as

Problem P2:

$$\min_{\Phi} \sum_{i=1}^N C_i W_i(\phi_i) \quad \text{s.t.} \quad g_1(\Phi) = c_1, \dots, g_n(\Phi) = c_n \quad (3)$$

where we replace the single normalization constraint by one or more constraints on the auxiliary variables which we obtain from the N equations of the form (2). From the theory of Lagrangian optimization, there are simple Kuhn-Tucker conditions that provide necessary conditions for optimality. If a Lagrange multiplier is associated with each constraint in P2, after eliminating the multipliers we obtain conditions which are typically of the form

$$\sum_{i=1}^L C_i \frac{\partial W_i(\phi_i)}{\partial \phi_i} = \sum_{k=1}^K C_k \frac{\partial W_k(\phi_k)}{\partial \phi_k} \quad (4)$$

for some $K, L < N$. In the absence of analytical expressions for $W_i(\phi_i)$, $i = 1, \dots, N$, we can use standard gradient-based optimization algorithms (hill-climbing or steepest-descent) [21, 8] to find the optimal Φ^* , provided the gradient in (4) is available. The corresponding Θ^* can then be obtained as a solution of the N linear equations (2). Alternatively, since the ϕ_i 's are themselves variables dependent on Θ , we may choose to control the θ_i 's: thus the minimization may be done

by adjusting Θ directly; an example will be given in the next section. The coupling between the transmission sets is now reflected in the adjustments of θ_i 's so as to achieve the derivative balance required by (4).

The preceding discussion motivates us to seek techniques for estimating $\frac{\partial W_i(\phi_i)}{\partial \phi_i}$ by simply observing the actual system in operation (or by simulating it). Along these lines, two approaches which have been proposed are Perturbation Analysis (PA) (e.g. [11, 14]) and the Likelihood Ratio (LR) methodology (e.g. [19, 12]). In this report, we apply PA techniques to estimate the sensitivities in question. The main idea in PA is to estimate $\frac{\partial W_i(\phi_i)}{\partial \phi_i}$ through the *sample derivative* $\frac{\partial L_i(\xi(\phi_i, \omega))}{\partial \phi_i}$, where $L_i(\xi(\phi_i, \omega))$ is a *sample function* obtained when a particular sample path $\xi(\phi_i, \omega)$ is observed, and $J_i(\phi_i) = E[L_i(\xi(\phi_i, \omega))]$. Here, ω is an element of the underlying probability space Ω , which is taken to be $[0, 1]^\infty$; thus, ω is simply viewed as a sequence of uniformly distributed random variables in $[0, 1]$. A sample derivative obtained through this approach is an unbiased estimate of $\frac{\partial W_i(\phi_i)}{\partial \phi_i}$ provided that [2]

$$\frac{\partial}{\partial \phi_i} E[L_i(\xi(\phi_i, \omega))] = E \frac{\partial}{\partial \phi_i} L_i(\xi(\phi_i, \omega)) \quad (5)$$

Since the form of the derivative estimator is the same for each transmission set, let us concentrate on an isolated transmission set. In particular, the queueing system corresponding to transmission set i behaves like a *modified G/D/1* queue with vacations, where vacations correspond to periods when the server is unavailable to transmission set i (i.e. the server is busy serving packets from other transmission sets) and service is synchronized to coincide with the start of a slot. That is, at the start of every interval of δ time units (i.e., each slot), the server is available to transmission set i and with probability $(1 - \phi_i)$ the server goes on vacation. A slot that is assigned to transmission set i is termed a '*transmission slot*', otherwise it is termed a '*vacation slot*'. We used the term *modified G/D/1* above, because an arriving customer must always wait until the end of the slot within which it arrives, even if the server is available (normally, this customer would immediately start service).

The key idea on which the estimator is based is to evaluate the effect of altering a given schedule (sequence of slot assignments) by converting one vacation slot to a transmission slot on an observed sample path. The vacation slot added to some hypothetical new schedule is termed a *phantom slot*. In presenting the estimation algorithm we define the following sample path quantities (based on a continuous-time model):

- K - Observation interval in number of busy periods
- N - Number of node i packet arrivals during observation interval
- $A_{i,j}$ - Arrival epoch of the i th packet in the j th busy period
- $T_{i,j}$ - Time instant when the i th transmission slot in the j th busy period begins
- $V_{i,j}$ - Time instant when the i th vacation slot in the j th busy period begins
- v_j - Number of vacation slots in the j th busy period.
- b_j - Number of packets in the j th busy period.

- $\beta_{ij} [\gamma_{ij}]$ - Packet index of the *first [last]* packet affected by the removal of the *ith* vacation slot in the *jth* busy period.

Figure 2 illustrates the aforementioned notation where vacation slots are shown darkened, arrivals are represented by arrows above the horizontal time axis, and departures by arrows below the time axis. In Figure 2, *phantomizing* (i.e., removal) of the 2nd vacation slot from the *mth* busy period introduces a service slot between the 1st and 2nd original service slots and causes all remaining customers in the busy period to begin service one service epoch earlier. Thus, the first customer affected by this additional service slot is the second arrival i.e., $\beta_{2m} = 2$. On the other hand, the last customer to be affected is $\gamma_{2m} = 3$. Finally, as a consequence of the phantom slot, the busy period in the nominal sample path has been split into two busy periods in the perturbed sample path.

We now state the following theorem (the proof can be found in [7]):

Theorem 1 *An unbiased estimate of $\frac{\partial W_i}{\partial \phi_i}$ is given by:*

$$\left[\frac{\partial W_i}{\partial \phi_i} \right]_{est} = \frac{1}{N(1 - \phi_i)} \sum_{j=1}^K \sum_{i=1}^{v_j} [V_{i,j} - T_{\gamma_{ij},j}]$$

where

$$\begin{aligned} \beta_{ij} &= \arg \max_m \{m : T_{m,j} < V_{i,j}\} + 1 \quad m = 1, \dots, b_j - 1 \\ \gamma_{ij} &= \arg \min_m \{m : A_{m+1,j} < T_{i,j}\} \quad m = \beta_{ij}, \dots, b_j \end{aligned}$$

and, by convention, $A_{b_j+1,j} \equiv A_{1,j+1}$

Observe that the first double summation (over the $V_{i,j}$ terms) is simply the sum of the vacation slot epochs, and in addition all remaining quantities are directly measurable from the sample path. An algorithm for implementing an estimator based on Theorem 1 can be found in the appendix of [7]. From an implementation viewpoint memory/computational overhead is minimized by noting that the majority of the sample-path quantities (in particular $T_{i,j}$ and $V_{i,j}$) need to be stored only for the duration of a busy period.

Finally, note that an estimator based on the dual process of converting a transmission slot into a vacation slot can also be similarly developed. We refer to transmission slots to be removed from the nominal system sample path as *marked* slots. Although *marking* (i.e., converting a transmission slot into a vacation slot) can be viewed as the dual process of *phantomizing*, there arise some fundamental differences between the *Phantom Slot* and *Marked Slot* estimator. Most notably, in the *Phantom Slot* estimator, perturbational effects are localized to the busy period within which they originate; as a result, implementation and analysis can be carried out for every busy period in isolation. Alternatively, in the *Marked Slot* estimator, the result of a perturbation due to marking a service slot in some busy period can propagate to subsequent busy periods and thus busy periods in the perturbed sample path may coalesce.

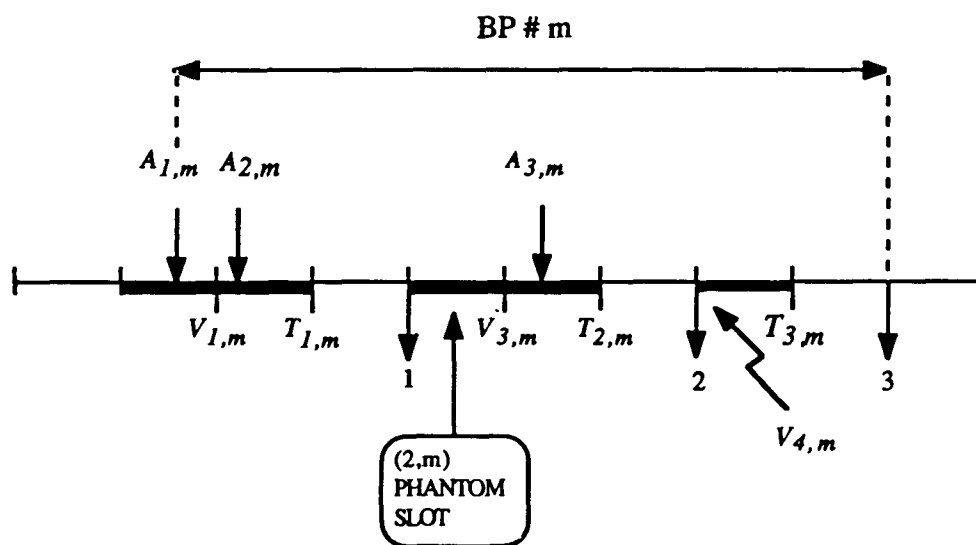


Figure 2a - Nominal Sample Path

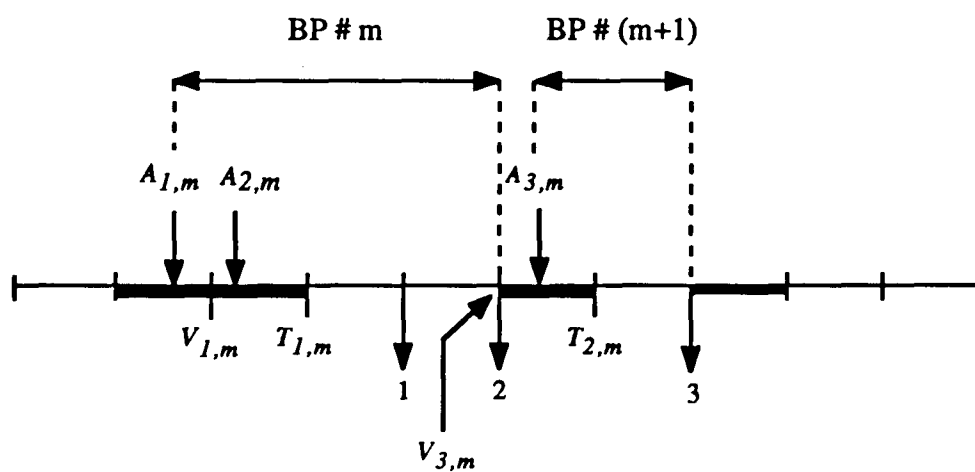


Figure 2b - (2,m) Phantom Slot Sample Path

3 A Schedule Optimization Algorithm

Now that we have at our disposal the derivative estimators required for solving the optimization problem (P2), we may proceed to develop a gradient-based algorithm for obtaining the optimal schedule (i.e., the optimal slot assignment probability vector $\Theta^* = [\theta_1, \dots, \theta_M]^T$ for a system with M transmission sets). In what follows, we limit ourselves to one such algorithm, similar to one used in [17], simply to illustrate our approach. Looking at (4), we see that our algorithm is such that it seeks to balance sums of derivatives. The main idea in our algorithm is the following: at every iteration, find the maximum imbalance condition and adjust the variables θ_i so as to reduce it.

We will proceed by considering two examples, one for the simple case where $\mathcal{C} = \mathcal{A}$, and one for the general case $\mathcal{C} \neq \mathcal{A}$. Finally, it should be clear that the sample-path-based nature of the derivative estimators on which our optimization approach relies possesses some inherent “adaptive” properties in the following sense: if the characteristics of various arrival processes change, then the derivative estimate also adjusts itself to reflect such a change from the observed data. The third example in this section is intended to illustrate these adaptive properties by allowing the topology of a system to change. For brevity we focus on the problem formulation and construction of the optimization schemes; simulation results and detailed discussions can be found in [7].

1. A Two-Node RN with $\mathcal{C} = \mathcal{A}$. In this example, we consider the transmission scheduling problem with two arrival streams and non-overlapping transmission sets. Our objective is to determine the optimal slot assignment probabilities so as to minimize the mean packet waiting time. For this simplified case, the optimization problem is reduced to balancing the derivatives of the individual packet waiting times. Let τ_k be the k th observation (iteration) interval over which we estimate the derivative $dW_i/d\theta_i$ and $\Theta^{(k)} = [\theta_1^{(k)} \ \theta_2^{(k)}]^T$ the probability vector over this interval⁵. At the end of the k th interval we compare individual node derivatives and set $i^* = \text{argmin}_i [dW_i/d\theta_i]$. The slot assignment probabilities are adjusted so as to increase the probability associated with i^* (and hence reduce the probability associated with the other arrival stream). In this simple example, we need only control a single probability. In particular, consider the control of θ_1 , and let the amount of change for the k th interval be δ_k where

$$\delta_k = \left| \frac{dW_1}{d\theta_1^{(k)}} - \frac{dW_2}{d\theta_2^{(k)}} \right| \times \eta_k$$

and η_k is a “step size” parameter⁶. At the end of the k th observation interval we update the node 1 probability as follows:

$$\theta_1^{(k+1)} = \begin{cases} \theta_1^{(k)} + \delta_k & \text{if } \frac{dW_2}{d\theta_2} > \frac{dW_1}{d\theta_1} \text{ and } \left(\frac{\lambda_2}{\theta_2^{(k+1)}} < 1 \right) \\ \theta_1^{(k)} - \delta_k & \text{if } \left(\frac{dW_2}{d\theta_2} < \frac{dW_1}{d\theta_1} \right) \text{ and } \left(\frac{\lambda_1}{\theta_1^{(k+1)}} < 1 \right) \\ \theta_1^{(k)} & \text{otherwise} \end{cases}$$

⁴When $\mathcal{C} = \mathcal{A}$ there is only one arrival stream in each transmission set.

⁵For this simple network θ_i is the probability that a transmission slot is assigned to arrival stream i .

⁶The step size sequence η_k must be properly chosen to guarantee convergence [8]. Intuitively, use of an excessively small step size will result in slow convergence, whereas use of an excessively large step size can result in oscillations and thus lack of convergence.

where a decrease in node 1 or node 2 probability is permitted *if and only if* at the new operating point $\Theta^{(k+1)}$ we do not violate the stability requirement. Finally, we assume a fixed step $\eta_k = 0.001$ and adjust the observation interval according to $\tau_k = kL_0$ with $L_0 = 100$ node 1 busy periods. The convergence of such an optimization scheme has been shown in [21].

2. A Four-Node RN with $\mathcal{C} \neq \mathcal{A}$. Consider the 4-node RN shown in Figure 1a where the objective is to minimize the average packet waiting time in the network. As was discussed in the introduction, we can translate this broadcast scheduling problem into a polling model with 3 overlapping transmission sets. Using the notation introduced in section 2, we have $\mathcal{Z}_1 = \{1\}$, $\mathcal{Z}_2 = \{1, 2\}$, $\mathcal{Z}_3 = \{2, 3\}$, $\mathcal{Z}_4 = \{3\}$. The optimization problem can be formulated as:

$$\min_{\Theta} \sum_{i=1}^4 f_i(\lambda_i) W_i(\Theta) \quad s.t. \quad \sum_{k=1}^3 \theta_k = 1 \quad (6)$$

where $\Theta = [\theta_1, \theta_2, \theta_3]^T$ and $f_i(\lambda_i)$ is the fraction of arrivals to the entire network that arrive at node i (typically a function of node arrival rate)⁷. Introducing the new variables ϕ_i , $i = 1, \dots, 4$, observe from the specification of the \mathcal{Z}_i sets that $\phi_1 = \theta_1$, $\phi_2 = \theta_1 + \theta_2$, $\phi_3 = \theta_2 + \theta_3$, and $\phi_4 = \theta_3$. Then, based on these variables and the normalization condition, we can reformulate the problem as:

$$\begin{aligned} \min_{\Theta} \quad & \sum_{i=1}^4 f_i(\lambda_i) W_i(\phi_i) \\ s.t. \quad & \phi_1 + \phi_3 = 1 \\ & \phi_2 + \phi_4 = 1 \end{aligned}$$

Using Lagrangian relaxation and eliminating Lagrange multipliers, we immediately obtain the following necessary conditions for an optimal point:

$$f_1(\lambda_1) \frac{\partial W_1}{\partial \phi_1} = f_3(\lambda_3) \frac{\partial W_3}{\partial \phi_3} \quad (7)$$

$$f_2(\lambda_2) \frac{\partial W_2}{\partial \phi_2} = f_4(\lambda_4) \frac{\partial W_4}{\partial \phi_4} \quad (8)$$

The optimization problem is then reduced to independently balancing two pairs of derivatives. We proceed as before by adopting a scheme so that at each iteration we minimize the maximum difference in the pairs of derivatives involved. Recall that the minimization is over the slot assignment probabilities, and, as before, let $\Theta^{(k)}$ be the slot assignment probability vector over the k th observation interval, where the duration of the latter is τ_k transmission set 1 busy periods. Over this interval each node i estimates its derivative $\frac{\partial W_i}{\partial \phi_i}$. At the end of the interval, there are two cases:

Case 1: If $\left| f_1 \frac{\partial W_1}{\partial \phi_1} - f_3 \frac{\partial W_3}{\partial \phi_3} \right| > \left| f_2 \frac{\partial W_2}{\partial \phi_2} - f_4 \frac{\partial W_4}{\partial \phi_4} \right|$ then we update the slot assignment proba-

⁷To simplify notation, we omit the dependence on λ_i when there is no ambiguity.

bilities as follows:

$$\Theta^{(k+1)} = \begin{cases} \theta_1^{(k)} - \Delta^k; \theta_3^{(k)} + \Delta^k & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} > f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} > f_4 \frac{\partial W_4}{\partial \phi_4} \right) \\ \theta_1^{(k)} - \Delta^k; \theta_2^{(k)} + \Delta^k & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} > f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} < f_4 \frac{\partial W_4}{\partial \phi_4} \right) \\ \theta_2^{(k)} - \Delta^k; \theta_1^{(k)} + \Delta^k/2; \theta_3^{(k)} + \Delta^k/2 & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} < f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} > f_4 \frac{\partial W_4}{\partial \phi_4} \right) \\ \theta_3^{(k)} - \Delta^k; \theta_1^{(k)} + \Delta^k & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} < f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} < f_4 \frac{\partial W_4}{\partial \phi_4} \right) \end{cases} \quad (9)$$

with

$$\Delta^k = \left| f_1 \frac{\partial W_1}{\partial \phi_1} - f_3 \frac{\partial W_3}{\partial \phi_3} \right| \times \eta_k$$

where η_k is the step size parameter. To ensure convergence, the step size sequence must be chosen to satisfy certain standard conditions (e.g., [8]). One family of sequences which satisfies these conditions is

$$\eta_k = \frac{1}{M k^p}$$

where M is a normalization constant and $0 \leq p \leq 1$. Finally, note that in (9) only components of Θ that change at the $(k+1)$ th iteration are explicitly shown; all remaining components are the same as in the prior iteration.

Case 2: If $\left| f_1 \frac{\partial W_1}{\partial \phi_1} - f_3 \frac{\partial W_3}{\partial \phi_3} \right| < \left| f_2 \frac{\partial W_2}{\partial \phi_2} - f_4 \frac{\partial W_4}{\partial \phi_4} \right|$ then:

$$\Theta^{(k+1)} = \begin{cases} \theta_1^{(k)} - \Delta^k; \theta_3^{(k)} + \Delta^k & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} > f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} > f_4 \frac{\partial W_4}{\partial \phi_4} \right) \\ \theta_2^{(k)} - \Delta^k; \theta_1^{(k)} + \Delta^k/2; \theta_3^{(k)} + \Delta^k/2 & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} < f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} > f_4 \frac{\partial W_4}{\partial \phi_4} \right) \\ \theta_3^{(k)} - \Delta^k; \theta_2^{(k)} + \Delta^k & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} > f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} < f_4 \frac{\partial W_4}{\partial \phi_4} \right) \\ \theta_3^{(k)} - \Delta^k; \theta_1^{(k)} + \Delta^k & \text{if } \left(f_1 \frac{\partial W_1}{\partial \phi_1} < f_3 \frac{\partial W_3}{\partial \phi_3} \right) \text{ and } \left(f_2 \frac{\partial W_2}{\partial \phi_2} < f_4 \frac{\partial W_4}{\partial \phi_4} \right) \end{cases} \quad (10)$$

with

$$\Delta^k = \left| f_2 \frac{\partial W_2}{\partial \phi_2} - f_4 \frac{\partial W_4}{\partial \phi_4} \right| \times \frac{1}{M k^p}$$

Finally, the length of the observation interval is controlled as follows:

$$\tau_k = k L_0 \quad (11)$$

where L_0 is an a priori chosen number of busy periods. The aforementioned optimization procedure incorporates both *step size* as well as *sample size control*⁸ [21]. Simulation results under asymmetric traffic loads can be found in [7].

⁸An alternative to controlling the step size sequence is to increase the estimation/observation horizon at each iteration. This is referred to as "sample size control." The convergence of such a scheme has been formally shown in [21]. An informal argument is as follows: If the estimators are consistent (which we conjecture to be the case), then as the observation interval increases, the sample derivative converges to the derivative of the expected waiting time.

3. Adaptivity properties when topology changes. In this example, we illustrate the adaptivity of our algorithm subject to a change in the topology of the RN considered in the last example. In particular, consider once again the 4-node RN in Figure 1a and assume that a new node wishes to participate in the transmission process. Moreover the “incoming” node (denoted as node 5) is located so that it is within the broadcast zone of nodes 1 and 2, i.e., node 5 is a neighbor of both nodes 1 and 2. The resultant topology is shown in Figure 3a by the inclusion of the dotted lines. The effect of the change in topology is to alter the transmission sets; the new transmission sets are now $\{(1, 2), (2, 3), (3, 4, 5)\}$ and the resultant polling model is shown in Figure 3b. The change in transmission sets gives rise to modified optimality equations, compared to the last example. In particular, the new Kuhn-Tucker optimality conditions are:

$$f_1(\lambda_1) \frac{\partial W_1}{\partial \phi_1} = f_3(\lambda_3) \frac{\partial W_3}{\partial \phi_3} \quad (12)$$

$$f_2(\lambda_2) \frac{\partial W_2}{\partial \phi_2} = f_4(\lambda_4) \frac{\partial W_4}{\partial \phi_4} + f_5(\lambda_5) \frac{\partial W_5}{\partial \phi_5} \quad (13)$$

A similar optimization scheme as that of example 2 can be now constructed to achieve the desired derivative balance. Again, a discussion of the simulation results can be found in [7].

4 A Deterministic Scheduling Policy

In practice, a randomized scheduling policy may not be desirable either because it involves a random number generation process or because it may be required that the time between successive server visits to a transmission set be bounded by a deterministic constant. In this section, we use our approach for obtaining optimal slot assignment probabilities in conjunction with the Golden Ratio (*GR*) policy presented in [15] to identify a deterministic scheduling policy with certain optimality properties. Our contribution is therefore to *optimize the Golden Ratio policy*. We then provide simulation results aimed at comparing this policy to several others, including our random polling approach developed in the previous sections. Note that in specifying a deterministic policy, we are seeking the specification of a scheduling *frame*: a frame consists of a fixed number F of time slots with each slot allocated to a given transmission set. The frame repeats itself in time, thus defining a periodic structure through which the server is assigned to each transmission set. The number F of slots in the frame is chosen according to a desired upper bound on transmission set intervisit times. The problem we address here is how to allocate the F slots to the M transmission sets.

The Golden Ratio Scheduling Policy. Let us briefly summarize this policy as presented in [15]. As before consider M transmission sets and let θ_i be the desirable fraction of slots to be assigned to transmission set i . Let $F_k = \frac{\varphi^k - (1-\varphi)^k}{\sqrt{5}}$ be the k th Fibonacci number, where $\varphi^{-1} = (\sqrt{5} - 1)/2 \approx 0.618034$. Then, let $N_k^i, i = 1, \dots, M$ be integers such that:

$$\lfloor \theta_i F_k \rfloor \leq N_k^i \leq \lceil \theta_i F_k \rceil \quad \text{and} \quad \sum_{i=1}^M N_k^i = F_k \quad (14)$$

with $N_k^0 \equiv 0$. Thus

$$\lim_{k \rightarrow \infty} \frac{N_k^i}{F_k} = \theta_i$$

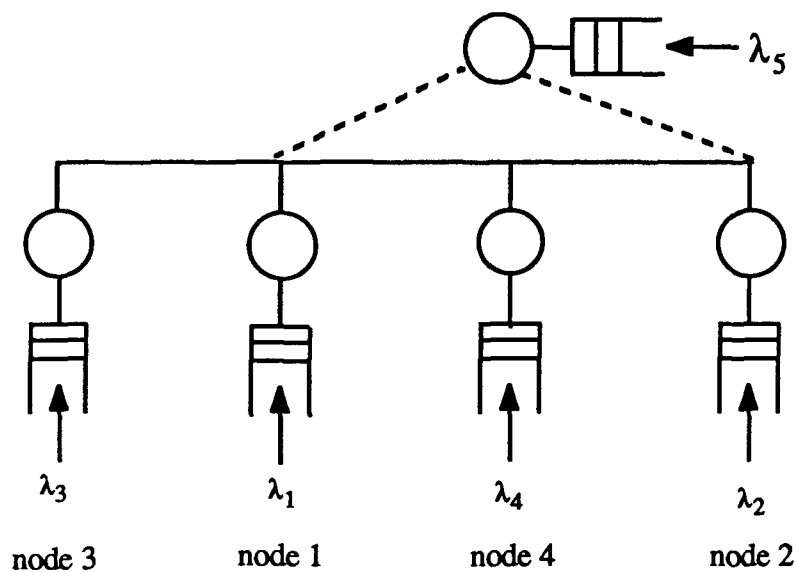


Figure 3a: RN Subject to a Topology Change

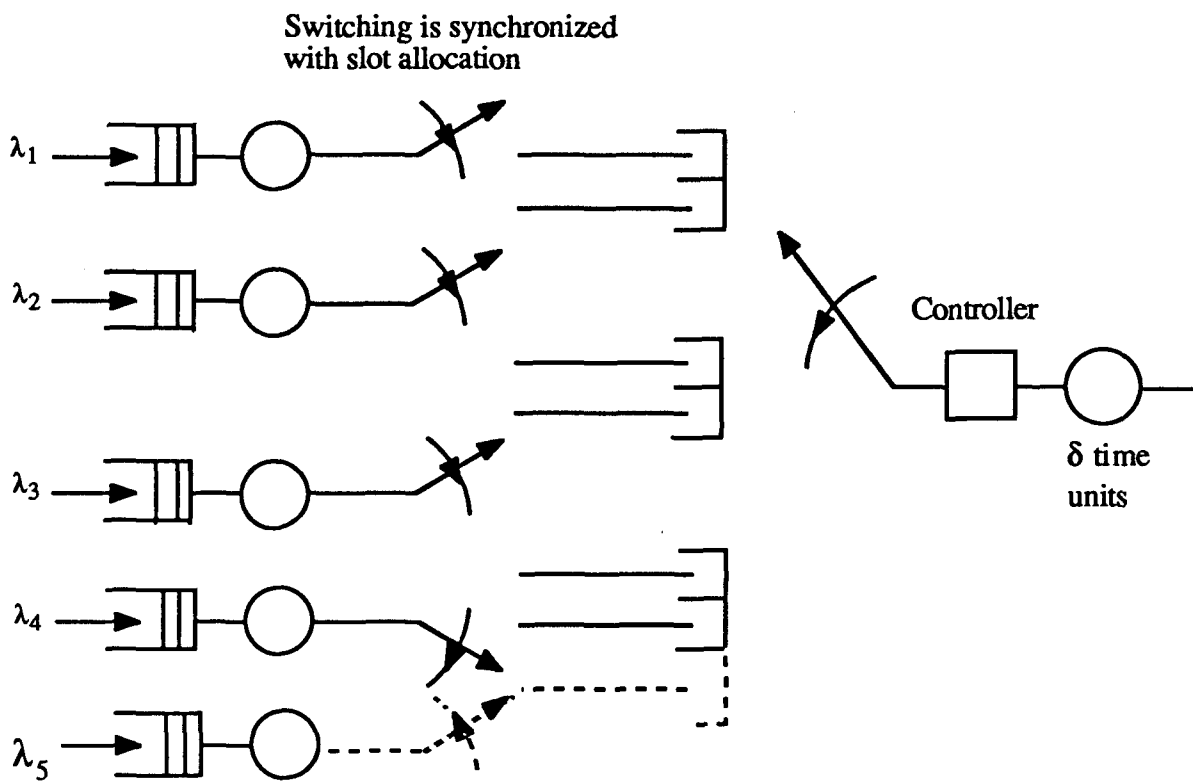


Figure 3b: Modified Multiclass Polling System

For each k the *GR* policy assigns N_k^i slots to transmission set i and attempts to distribute the allocation of slots *uniformly* over a frame of size F_k . Note that although in the limit $k \rightarrow \infty$ visits to transmission set i are equally spaced, because of the generally overlapping transmission set, visits to a particular node need not be equally spaced.

Let $\text{frac}(x) = x - \lfloor x \rfloor$, $a_j = \text{frac}(j\varphi^{-1})$ and $A_{F_k} = \{a_j | j = 0, \dots, F_k - 1\}$, where the n th ($1 \leq n \leq F_k$) smallest element of A_{F_k} is associated with the n th slot in the frame. The final step in the *GR* policy is then one which assigns to class i the following slots within a frame:

$$\left\{ a_j \mid \sum_{m=1}^{i-1} N_k^m \leq j < \sum_{m=1}^i N_k^m \right\} \quad (15)$$

Note that k determines the desired length of the scheduling frame, which we assume is given. Our task is to specify the optimal θ_i , $i = 1, \dots, M$. Once this is done, one can use (14) and (15) to allocate slots in the frame. Finally, observe that if all θ_i are equal then *GR* reverts to the simple Round Robin policy discussed next. We illustrate the construction of the *GR* policy through the following example.

Example: To illustrate the procedure consider the *GR* policy corresponding to the optimal slot allocation probabilities $\theta_1 = 0.32$; $\theta_2 = 0.18$ and $\theta_3 = 0.50$. Let $k = 7$ so that we define a frame of length $F_7 = 13$, and, using these values of θ_1 , θ_2 , θ_3 in (14), we get $N_7^1 = 4$, $N_7^2 = 3$ and $N_7^3 = 6$. From (15) the slots corresponding to $\{0, \varphi^{-1}, \text{frac}(2\varphi^{-1}), \text{frac}(3\varphi^{-1})\}$ are assigned to class 1; the slots corresponding to $\{\text{frac}(4\varphi^{-1}), \dots, \text{frac}(6\varphi^{-1})\}$ are assigned to class 2 and, lastly, the slots corresponding to $\{\text{frac}(7\varphi^{-1}), \dots, \text{frac}(12\varphi^{-1})\}$ are assigned to class 3. Consequently we can write $A_{13} = \{0, \varphi^{-1}, \text{frac}(2\varphi^{-1}), \dots, \text{frac}(12\varphi^{-1})\}$. Evaluating the individual terms in A_{13} and rearranging then in an increasing order (recalling that the n th smallest element corresponds to the n th slot) we obtain the following polling sequence: $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 3 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 3$.

The Round Robin Policy In the Round Robin (*RR*) policy, assuming M transmission sets, slots are assigned to each transmission set according to the following simple sequence: $1 \rightarrow 2 \rightarrow \dots \rightarrow M \rightarrow 1 \rightarrow 2 \rightarrow \dots \rightarrow M \dots$. If we define a frame of length M slots, then *RR* assigns each transmission set exactly 1 slot in every M slots. *RR* scheduling policies in the context of polling system have been well researched and have been shown to be optimal (over the class of all admissible policies) for systems with symmetric traffic and non-overlapping transmission sets [22, 15]. We conjecture a similar result for overlapping transmission sets. In particular the superiority of *RR* over random polling, under symmetric traffic, can be attributed to the higher variance of the intervisit time (time, in slots, between successive visits to a transmission set) in random polling. However as we deviate from symmetric traffic conditions, *RR* rapidly deteriorates in performance, whereas random polling is fairly 'robust.' The last observation is true since random polling will adapt to the changing traffic and allocate slots to reflect the change. The optimized *GR* policy may be thought of as an effort to combine the variance properties of the frame structure with the adaptive nature of random polling in order to design a frame which reflects the optimal slot allocation probabilities.

The Maximal Traffic Scheduling Policy The *Maximal Traffic Scheduling Policy* (*MTP*) is a variant to the optimal *GR* policy where, rather than using the optimal θ_i 's to construct the frame, we construct a frame such that the frequency of visits to a particular transmission set is

a reflection of the ‘maximal traffic’ at that transmission set. Equivalently *MTP* assigns priorities based on the maximum arrival rate to a transmission set. For example, if in Figure 1b we let the arrival rates be $\lambda_1 = 0.1, \lambda_2 = 0.2, \lambda_3 = 0.3, \lambda_4 = 0.2$, then the maximal traffic at transmission set 1 is $\max(\lambda_1, \lambda_2) = 0.2$; similarly for transmission sets 2 and 3 the maximal traffic is 0.3. Consequently the desirable fraction of slots to be assigned to transmission sets 1, 2 and 3 is $\frac{2}{8}, \frac{3}{8}$ and $\frac{3}{8}$ respectively. Using the same procedure as for the *GR* (see (15)) the *MTP* polling sequence is: $1 \rightarrow 3 \rightarrow 2 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 2$.

MTP is an inherently conservative policy in that it ignores the overlapping structure and allocates slots assuming that if an arrival stream belongs to a particular transmission set, then *all* packets from that stream are served in that transmission set. The efficiency of *MTP* is contingent on the arrival rates of overlapping arrival streams (streams that belong to more than one transmission set). For example, if the maximal arrival rate in each transmission set corresponds to a non-overlapping stream, then *MTP* incurs no penalty by ignoring the overlapping structure. Alternatively, if the maximal rate corresponds to an overlapping stream, then *MTP* assigns surplus slots to that transmission set at the expense of increasing the mean intervisit time for streams not belonging to that set. Finally, if the maximal arrival rate is the same for each transmission set then *MTP* reverts to *RR*.

A comparison of random polling (as developed in previous sections), *RR*, optimal *GR* (where the optimization is based on the slot assignment probability vector Θ^* determined through our approach), and *MTP* is presented in [7] for various system parameter (arrival rates and distributions). The conclusion drawn is that in each case *GR* performs better than *RR* or random polling; however random polling, even under optimal slot assignments, does not always perform better than *RR*. The last observation is attributed to the higher variance of the intervisit time in random polling.

5 Scheduling in Networks with Voice Traffic

For the remainder of the report we focus on scheduling voice calls in an N -node network with (for simplicity) N non-overlapping (distinct) transmission sets. The analysis readily extends to the case with overlapping transmission sets. In this model, ‘transmission set’ is synonymous with ‘node’ and thus, for brevity, the latter terminology is adopted. Recall that voice traffic is *delay-sensitive traffic* in the following sense: voice packets cannot be queued for more than n time slots, where n is fixed. Thus, upon arrival, a voice call is accepted if it can be assigned a transmission slot within the *call setup time*⁹ otherwise it is *blocked*. Once a call is accepted, then due to the cyclic nature of the polling policy, we can guarantee that packets belonging to this call are assigned a transmission slot every n time slots. We consider a *cyclic policy*, where the polling cycle is specified by a *frame* comprised of n slots, such that a slot is uniquely assigned to a particular transmission set. Thus, a frame is characterized by a vector $[c_1, \dots, c_n]$ where $c_i \in \{1, \dots, N\}$ is the transmission set (node) assigned to slot i . Our objective then is to determine the optimal frame that minimizes the average call blocking probability.

The delay-sensitive blocking model can be used to model *voice call* processing in an N -node multihop RN [10, 24]. When processing voice traffic with *circuit switching*, a voice call arriving

⁹In this report the call setup time is defined as the maximum duration a call can be queued before a decision to accept/reject the call is made.

at a node reserves (for the duration of the voice call) the desired bandwidth at each node along a multihop path. If, upon arrival of a voice call, bandwidth is unavailable at any of the intermediate nodes required, the call is blocked. This is also referred to as 'voice admission control', a problem also considered in [1] using a model not based on slotted time, but rather frequency division multiplexing to overcome the problem of channel access conflicts. In the context of our slotted-time model, *Time Division Multiplexing (TDM)* is employed to provide contention-free multiplexing of the bandwidth. Moreover, the duration of a time slot is determined by the bandwidth requirement of each node (for simplicity, we assume that the bandwidth requirements are the same at all nodes).

In the general *TDM* framework, a *frame* consisting of n time slots is defined so that each slot is assigned to a voice call (or several voice calls belonging to the same transmission set). The size of the frame (i.e., the number of slots n) is chosen so that proper voice transmission is ensured. The frame is repeated in time so that the controller assigns the server to each transmission set in accordance with the frame structure. Thus, a call is blocked if it cannot be assigned a slot in the *TDM* frame. The design of the frame then represents a means of controlling the performance of the system in terms of the blocking probability of calls. In the remainder of this section we describe the call arrival process in more detail, and define two modes of delay-sensitive operation, termed *slot delay blocking* and *frame delay blocking*.

We refrain from providing a specific model for voice; rather, we assume that sampling and encoding are such that at most one packet is submitted for transmission per frame by any individual call. This guarantees that if a call is accepted, then no packet from that call will be queued for service. We can now describe our model as follows. At each node i , $i = 1, \dots, N$, there is a process characterizing voice call arrivals. We assume an arbitrary interarrival distribution for calls, and allow for correlations between call arrivals. The j th call at node i is characterized by the pair (A_j^i, η_j^i) , where A_j^i is the call arrival epoch and η_j^i is the call duration, specified as a number of frames. The discrete random variable η_j^i has an arbitrary distribution. Equivalently η_j^i can be viewed as the number of packets in the j th node i call. Note that A_j^i and η_j^i do not have to be known *a priori*. The decision to accept or reject a call is made independent of the call duration, and moreover, once the call is accepted, it is guaranteed one slot per frame for the duration of the call.

Next, we describe the operation of the system from node i 's point of view; the operation is the same from the point of view of any other node. The server is assigned to nodes based on a frame consisting of n slots (as already mentioned, n depends on the characteristics of the delay-sensitive traffic we are serving). Each slot is assigned to some node. Thus, from node i 's perspective, the frame consists of slots assigned to i (*transmission slots*) and slots assigned to some other node (*vacation slots*). Suppose a frame is such that n_i slots are assigned to node i . For any time instant, let f_i , $0 \leq f_i \leq n_i$, be the number of transmission slots not committed to any ongoing node i call. Then, when a call (A_j^i, η_j^i) is submitted to the system, the call is admitted if $f_i > 0$, in which case f_i is decremented by 1; otherwise, the call is blocked and considered lost. Moreover, this call reserves the slot it is assigned for the next η_j^i frames, i.e., if the j th call is accepted and begins using the k th slot of the m th frame on the time line, then it reserves the k th slot for the $\{(m+1), \dots, (m+\eta_j^i-1)\}$ frames. These slots are now considered *unavailable* to future voice calls arriving at i .

The model is complete once we specify the call setup time, i.e., how long an arriving call can wait for an accept/reject decision. We consider two variants on the call setup time. In the first variant, henceforth referred to as *Slot Delay (SD) blocking*, an arriving call is queued until the

beginning of the next slot (vacation or transmission) only, resulting in a call set up of one time slot. A call is therefore blocked if within one slot duration of the call arrival epoch an *available* transmission slot is not assigned to node i (recall that a transmission slot is deemed available if it has not been reserved by any prior voice call). This is true regardless of whether there are any transmission slots actually available in the frame. If the call is accepted, it proceeds to reserve slots in subsequent frames as described in the preceding paragraph.

Alternatively, in *Frame Delay (FD) blocking*, the decision to accept calls is made at the beginning of each frame (for a maximum call setup of $2n - 1$ slot durations¹⁰); all calls that arrive during a frame are therefore queued up to the beginning of the next frame. At that time, those calls that cannot be assigned an available transmission slot are blocked. As before, if call (A_j^i, η_j^i) is accepted, it reserves the slot it is assigned for η_j^i frames. Note that this model allows for an arbitrary selection of which calls at the beginning of any frame will be blocked, whereas in *SD* queuing the *First-In-First-Out* property is always preserved. It is also worth pointing out that, in *FD* blocking, if there are $f_i > 0$ available transmission slots when a call arrives, the precise slot identity in the frame which is assigned to this call does not affect the performance of the system in terms of future blocked calls.

6 Discrete Optimization Problem Formulation

We now formulate the scheduling problem as an optimal slot allocation problem. Let c_j denote the node assigned to slot j in a frame consisting of n slots. Then, let S be the finite set of feasible frames where:

$$S = \{[c_1, \dots, c_n] : c_j \in \{1, \dots, N\}\}$$

By "feasible" we mean that the frame must be chosen to satisfy some basic requirements, such as that at least one slot is assigned to each node. For any $s \in S$, let $J_i(s) = E[L_i(s)]$ be the node i cost function, where as before $L_i(s)$ is the sample node i cost. Then, our objective is to select $s \in S$ so as to solve the following discrete optimization problem:

Problem P3:

$$\min_{s \in S} \sum_{i=1}^N \beta_i E[L_i(s)]$$

where β_i is the weight associated with node i . This is generally a difficult problem, as it involves both optimization over a discrete set and estimation of the total cost under all possible frames $s \in S$. Rather than attempting to solve it, we transform (P3) into a *continuous parameter optimization problem* as follows.

6.1 Transforming the Discrete Optimization Problem into a Continuous Optimization Problem

Let $\theta_i = P[c_j = i]$, independent of the slot index $j = 1, \dots, n$. We can then construct a frame through a randomization mechanism based on a probability vector $\theta = [\theta_1, \dots, \theta_N]$, with $\sum_{i=1}^N \theta_i =$

¹⁰The maximum call setup corresponds to a voice call that arrives just after the beginning of a frame and is assigned the last time slot in the subsequent frame.

1, whereby each slot is independently assigned to node i with probability θ_i . In this setting, θ represents the control parameter which can be optimally selected so as to minimize the desired cost function. In particular, we replace problem (P3) by the following constrained continuous parameter optimization problem:

Problem P4:

$$\min_{\theta} \sum_{i=1}^N \beta_i E[L_i(\theta)] \quad \text{s.t.} \quad \sum_{i=1}^N \theta_i = 1$$

The relationship between the slot assignment probability vector θ and the actual frame s can be made precise as follows. Let $\mathbf{r} = [r_1, \dots, r_n]$ be an n -dimensional vector of independently generated random numbers with every r_j uniformly distributed in $[0, 1]$. A frame $\mathbf{s}(\theta, \mathbf{r}) = [c_1, \dots, c_n]$ is obtained using the standard inverse transform technique for generating random variates c_1, \dots, c_n as follows [16]:

$$c_j = \min\{i : r_j \leq \sum_{k=1}^i \theta_k, \quad i = 1, \dots, N\} \quad (16)$$

It is then clear that, for a given \mathbf{r} , by adjusting the parameters θ_i , $i = 1, \dots, N$, we can affect c_j , $j = 1, \dots, n$, and, consequently, control the design of the frame. From a notational standpoint, it will be useful to distinguish between a frame generated through (16), which is denoted by $\mathbf{s}(\theta, \mathbf{r})$, and a frame denoted by \mathbf{s} , which is simply an element of the set S .

The obvious advantage of replacing the discrete optimization problem (P3) by the continuous optimization problem (P4) is the fact that the latter may be solved through standard nonlinear programming techniques (e.g., using a Lagrangian relaxation approach). Moreover, if closed-form expressions for $J_i(\cdot)$ are available, (P2) is reduced to solving a system of algebraic equations. If, on the other hand, such expressions are not available, then we can still resort to standard gradient-based optimization algorithms (hill-climbing or steepest descent). Thus we need to derive estimators similar to those developed in [7] and in section 2.

Since our goal then is to estimate $\partial E[L_i(\theta)]/\partial \theta_i$ from a single sample path, we need to generate a sample path of RN operation in a stochastic environment. To do so, we need to specify a frame $\mathbf{s}(\theta, \mathbf{r})$. Assuming that a control vector θ is specified, then a sample path is characterized by the vector \mathbf{r} and we write in (P4) $E[L_i(\theta)] = E[L_i(\mathbf{s}(\theta, \mathbf{r}))]$. Consequently, we see that the solution of (P4) is dependent on the initial selection of \mathbf{r} , which is subsequently held fixed throughout the optimization procedure. If we repeat the process with a new initial \mathbf{r} , then, in general, a different optimal point θ^* is attained. A solution to (P4) is therefore characterized by the pair $(\theta_k^*, \mathbf{r}_k)$ where the subscript refers to the k th optimization run. However, we claim that, the frame $\mathbf{s}(\theta_k^*, \mathbf{r}_k)$ is fixed for all k , and, under certain conditions, it is also the solution to (P3).

In the next section we adopt the *Marked Slot* methodology proposed in [4] in order to obtain an unbiased estimate of $\frac{\partial J_i}{\partial \theta_i}$ for any node $i = 1, \dots, N$. Note that $J_i(\cdot)$ is a function of θ_i only, i.e., the fraction of calls blocked at node i depends only on the fraction of slots assigned to i . We can therefore concentrate on an isolated node i . Recall that a frame, from node i 's perspective, is comprised of transmission and vacation slots; therefore a frame $\mathbf{s} = [c_1, \dots, c_n]$ can be represented

by the sequence $\{u_j\}_{j=1}^n$ where the Boolean variable u_j is given by

$$u_j = \begin{cases} 1 & \text{if the } j\text{th slot in frame } s \text{ is a transmission slot for node } i \\ 0 & \text{otherwise} \end{cases}$$

For some given θ and \mathbf{r} we can construct s through the following:

$$u_j = \begin{cases} 1 & \text{if } c_j = i \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

where c_j is given by (16). Alternatively, since from node i 's point of view slot j in the frame is a transmission slot with probability θ_i and a vacation slot with probability $(1 - \theta_i)$, we have

$$u_j = \begin{cases} 1 & \text{if } r'_j \leq \theta_i \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

where $\mathbf{r}' = [r'_1, \dots, r'_n]$ is a vector of independently generated random numbers, each uniformly distributed in $[0, 1]$, which can be obtained from the original vector \mathbf{r} through a simple linear transformation.

7 Derivative Estimation through the Marked Slot Approach

In this section we use the concept of *marking* (as defined earlier; see section 2) transmission slots in order to obtain unbiased derivative estimates based on observation of a single sample path of our system. The fundamental difference between marking a slot in a purely random policy as opposed to a frame-based one is that in the latter the marking of a slot in a given frame *also results in the marking of the same slot in every subsequent frame*. We refer to the *j*-marked slot system as the sample path generated when the *j*th transmission slot in the nominal frame is marked. Let us now concentrate on the specific performance measure of interest to our problem, i.e., let $J_i(\theta_i)$ be the *expected fraction of calls blocked at node i* after K call arrivals, which we will denote by P_i . Define the following sample path quantities for each node i :

- $b_i[j]$ - Total number of blocked calls at node i in the nominal [*j*-marked slot] system
- $\alpha_i(j)$ - Total number of node i calls *accepted* in the *j*th transmission slot of any frame.
- $a_i(k)$ - Number of call arrivals at node i during the *k*th frame
- $d_i(k)$ - Number of call completions at node i during the *k*th frame
- $f_i(k)$ - Number of transmission slots available to node i in the *k*th frame (i.e., slots assigned to i which are not currently used by a node i call)

Then, we consider the sample function $L(\theta_i, \omega) = b_i/K$, with $P_i = E[b_i]/K$. Note that we limit ourselves here to sample paths defined by K call arrivals at node i . Under standard ergodicity conditions, we expect that as $K \rightarrow \infty$, L converges a.s. to the steady-state blocking probability at node i . With the notation defined above we have the following theorem (the proof can be found in [5]):

Theorem 2 *An unbiased estimate of $\frac{dP_i}{d\theta_i}$ is given by:*

$$\left[\frac{dP_i}{d\theta_i} \right]_{est} = \frac{1}{K\theta_i} \sum_{j=1}^{|A|} [b_i - b_i(j)] \quad (19)$$

In summary, (19) requires us to calculate the change in performance when the j th (for all $j \in A$) transmission slot is marked in the nominal frame. In the remainder of this section we present expressions for evaluating this change, in terms of quantities directly observable on the nominal sample path, first operating under *SD* blocking and then *FD* blocking. Again, for brevity, we will present only the final result and omit the actual derivation.

7.1 Derivative Estimation with Slot Delay Blocking

Recall, that the effect of marking the j th slot in the first frame is to mark the j th slot in every subsequent frame. The consequences of such a *coupled marking* is that all calls that were originally accepted in the j th transmission slot of *any* frame in the nominal system, will now be blocked in the j -marked system. Therefore, the change in number of node i blocked calls when the j th transmission slot is marked is given by $\alpha_i(j)$ or the total number of node i calls *accepted* in the j th transmission slot of any frame. Thus we have the following result:

Theorem 3 *An unbiased estimate of $\frac{dP_i}{d\theta_i}$ is given by:*

$$\left[\frac{dP_i}{d\theta_i} \right]_{est} = \frac{b_i/K - 1}{\theta_i} \quad (20)$$

The proof can be found in [5].

Note that since the estimator in (20) is unbiased, it follows that

$$\begin{aligned} \frac{dP_i}{d\theta_i} &= E \left[\frac{b_i/K - 1}{\theta_i} \right] \\ &= \frac{P_i - 1}{\theta_i} \end{aligned} \quad (21)$$

The first-order differential equation in (21) can be solved to obtain the linear equation for P_i :

$$P_i = \hat{m}\theta_i + 1 \quad (22)$$

where \hat{m} is the slope, an unbiased estimate of which is given by (20). The ramification of this result is that (P4) is now transformed into a linear programming problem and can be efficiently solved. It is interesting to note that this result is easily derived through the “perturbation-like” argument given above, which is completely independent of the call arrival process or the distribution characterizing call durations. It is an example of a simple sample-path-type argument that can be used not to derive explicit derivative estimates, but to establish a functional relationship between a performance metric, P_i , and a parameter of interest, θ_i .

7.2 Derivative Estimation with Frame Delay Blocking

The primary difference between slot marking with *SD* and *FD* blocking is that, in the latter, marking a transmission slot does not necessarily imply that a call, if accepted in the marked slot, will be blocked in the perturbed system. Define a “tagged” call to be a call that is accepted in the nominal but not in some j -marked sample path. Then, by definition, $\Delta b_i(j)$ is given by the total number of tagged calls over the observation interval. Therefore if we can monitor the *tagging* process then we can derive the appropriate estimator. Before proceeding further we define

$$n_i = \sum_{j=1}^K \mathbf{1}[s_j = i] \quad i = 1, \dots, N$$

where $\mathbf{1}(\cdot)$ is the standard indicator function. Thus, n_i is simply the number of slots allocated to node i in some frame s . We shall now make the following simplifying assumption for our problem:

- *A1*: $L_i(s)$ depends only on the number of slots assigned to node i , i.e., $L_i(s) = L_i(n_i)$.

Clearly, *A1* is justified when the sample function is the blocking probability under *FD* blocking. However, the assumption would not hold in cases where the identity of the slots assigned to node i affects that node's cost function. For example, if the sample function were the individual node mean waiting time, then clearly the latter is a function of the position of the transmission slots within the frame, and not just of the number of such transmission slots. Lastly, note that as a consequence of *A1* there can be at most one tagged call in the nominal system at any instant.

To formalize the process of call tagging, we define the binary variable $z_i(k)$, where we use $z_i(k) = 1$ to denote the fact that there is a tagged call at the beginning of the k th frame. The dynamics of the tagging process are then given by:

$$z_i(k+1) = \begin{cases} 1 & \text{if } [a_i(k) \geq f_i(k) + d_i(k)] \text{ and} \\ & [(z_i(k) = 0) \text{ or (tagged call departs in frame } k)] \\ 0 & \text{if (tagged call departs in frame } k) \text{ and } [a_i(k) < f_i(k) + d_i(k)] \\ z_i(k) & \text{otherwise} \end{cases}$$

with initial condition $z_i(1) = 0$. We claim that (for details see [5]):

Theorem 4 *An unbiased estimate of $\frac{dP_i}{d\theta_i}$ is given by:*

$$\left[\frac{dP_i}{d\theta_i} \right]_{est} = \frac{-|A|}{K\theta_i} \sum_{k=1}^F \mathbf{1}[a_i(k) \geq f_i(k) + d_i(k)] \mathbf{1}[(z_i(k) = 0) \text{ or (tagged call departs)}]$$

where F is the number of frames contained in a sample path defined by K call arrivals at node i .

Finally, a reduced-variance estimator can be obtained by exploiting the fact that $\theta_i = E|A|/n$. Thus, the estimator we finally use (which, in general, will no longer be unbiased) is

$$\left[\frac{dP_i}{d\theta_i} \right]_{est} = -\frac{n}{K} \sum_{k=1}^F \mathbf{1}[a_i(k) \geq f_i(k) + d_i(k)] \mathbf{1}[(z_i(k) = 0) \text{ or (tagged call departs)}] \quad (23)$$

For the remainder of the report we focus on the optimization problem assuming *FD* blocking.

8 Convergence to a Global Optimum

Before presenting our main result, let us summarize our approach by means of the following optimization scheme (S1). For brevity, we denote the estimate in (23) by D_i .

1. Initially select some \mathbf{r} and $\theta^{(0)}$. Hence, determine an initial frame $\mathbf{s}^{(0)} = \mathbf{s}(\theta^{(0)}, \mathbf{r})$ through (16).
2. Observe a sample path under $\mathbf{s}(\theta^{(0)}, \mathbf{r})$ and estimate the derivatives $D_i^{(0)}$ for all $i = 1, \dots, N$.
3. For any $m = 0, 1, \dots$, iterate as follows:

$$\theta_i^{(m+1)} = \theta_i^{(m)} + \eta_m D_i^{(m)} \quad (24)$$

$$c_j^{(m+1)} = \min\{i : r_j \leq \sum_{k=1}^i \theta_k^{(m+1)}, i = 1, \dots, N\} \quad (25)$$

with some provisions to ensure that $\sum_{i=1}^N \theta_i^{(m)} = 1$ for all $m = 0, 1, \dots$.

4. Observe a sample path under $\mathbf{s}(\theta^{(m)}, \mathbf{r})$ to estimate the derivatives $D_i^{(m)}$ for all $i = 1, \dots, N$, and repeat step 3 until we satisfy *stopping condition*.

Of course, (24) is a standard stochastic approximation scheme driven by the derivative estimates D_i^m with $\{\eta_m\}$ an appropriately selected step size sequence. There are several important observations: first after an update in the auxiliary vector $\theta^{(m)}$, *the frame is also updated* using the transformation defined in (16). Second, the iterative scheme is terminated when we reach a stopping condition, and not necessarily when θ_i^m 's have converged. In particular, the stopping condition is defined as one where

$$\max\{\tilde{D}_i^{(m)}, i = 1, \dots, N\}$$

is minimized over the iteration index m . Finally, note that \mathbf{r} remains fixed throughout the process. For more details on the algorithm the reader is referred to [6].

In addition, we make the following assumption regarding the cost functions of interest:

- A2: For all $i = 1, \dots, N$, $L_i(n_i)$ is such that $L_i(n_i + 1) - L_i(n_i) > L_i(n_i) - L_i(n_i - 1)$.

We can now establish the following result. Let us first consider a specific sample path and apply the optimization scheme (S1). Let θ^s be the probability vector when the stopping condition is met in (24), and let $\mathbf{s}(\theta^s, \mathbf{r})$ be the corresponding frame. This frame is an optimal one in the discrete space S , i.e., $\mathbf{s}(\theta^s, \mathbf{r}) = \mathbf{s}^*$, where \mathbf{s}^* is independent of \mathbf{r} and of the sequence $\{\theta^{(m)}\}$, $m = 0, 1, \dots$ which leads to the stopping point in (24).

Theorem 5 *Under assumptions A1-A2, the optimization scheme (S2) applied to a particular sample path of the underlying system yields a frame which is optimal in the discrete space S .*

The proof can be found in [6].

We next present two applications of this approach to voice call processing in RNs. We first consider the optimal frame design to minimize blocking in a fully connected RN. In the second application, we consider the optimal admission control problem in a multihop RN.

9 Optimal TDM Frame Design

This application is motivated by the need to schedule voice call transmissions (broadcast mode) in an N -node Radio Network [24]. In particular consider a 3-node RN and for simplicity, assume that all nodes are neighbors of every other node i.e., the network is fully connected. The stochastic discrete optimization problem is then to select the optimal frame \mathbf{s}^* to minimize $\sum_{i=1}^3 \beta_i P_i(\mathbf{s})$ where $P_i(\mathbf{s})$ is the node i blocking probability operating under FD blocking.

We employ a gradient based optimization algorithm (see [17]), where the parameters of interest are: (a) the observation interval τ_k (specified in terms of the number of calls arrivals) over which the derivatives $dP_i/d\theta_i$ are estimated, and (b) the step size η_k which is used to control the adjustment at the k th update. In the algorithm considered here we assume a constant step size $\eta = 1.0$ and increasing observation interval lengths. In particular, $\tau_k = \tau_{k-1} + C$ with $\tau_1 = 1000$ and $C = 200$ node 1 call arrivals. At the end of the k th observation interval we compare individual node derivatives and set $i^* = \arg \min_i [dP_i/d\theta_i]$ and $j^* = \arg \max_i [dP_i/d\theta_i]$. The slot assignment probabilities are adjusted so as to increase the probability associated with i^* and reduce the probability associated with j^* .

In the simulation results shown below, we assume a $T1$ frame with $n = 24$ slots, and a symmetrically loaded system with Poisson arrivals ($\lambda_i = 0.2 \forall i = 1, \dots, 3$) and fixed call durations of 5 frames. Thus, even though this system is difficult to analyze, it follows from the symmetry that the optimal frame is $[8, 8, 8]$. We considered three single-run optimization experiments (with different randomization vectors $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$). In each case, the initial slot assignment probability vector is $\theta^{(0)} = [0.7, 0.15, 0.15]$. Table 1 shows the initial and final frames, as well as the final slot assignment probabilities, where the latter two are indexed by x , the number of iterations on θ until we first reach the optimal frame. After this point we oscillate between the optimal and one of its neighbors. The oscillatory behavior is typical of most gradient-based iterative schemes and in practice it is easy to identify this situation and select the true optimal allocation.

<i>Exp.</i>	$\mathbf{s}^{(0)}$	\mathbf{r}	$\mathbf{s}^{(x)}$	$\theta^{(x)}$	x
1	(18, 3, 3)	\mathbf{r}_3	(8, 8, 8)	[0.391, 0.261, 0.348]	29
2	(20, 3, 1)	\mathbf{r}_1	(8, 8, 8)	[0.292, 0.197, 0.511]	30
3	(21, 1, 2)	\mathbf{r}_2	(8, 8, 8)	[0.263, 0.185, 0.552]	38

Table 1: Frame Optimization for a 3-node RN

10 Optimal Admission Control in Blocking Networks:

In this application we determine the optimal admission control policy to minimize blocking in networks with fixed routing (specified as Source/Destination pairs). Each S/D pair can be viewed as a, typically multihop, circuit where the i th such circuit is denoted by c_i . We consider a 4-node network with 5 S/D pairs (see Figure 4a). We assume homogenous resources at each node, in particular each node in the network has n transceivers, and such that a transceiver can be used for simultaneously transmitting and receiving a packet. In tandem queue networks (see Figure

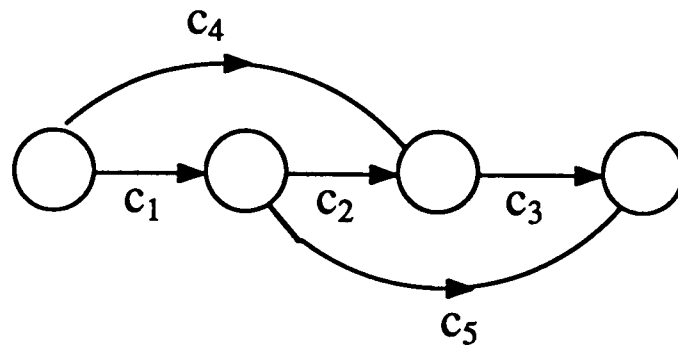


Figure 4a: 4-Node Multihop RN

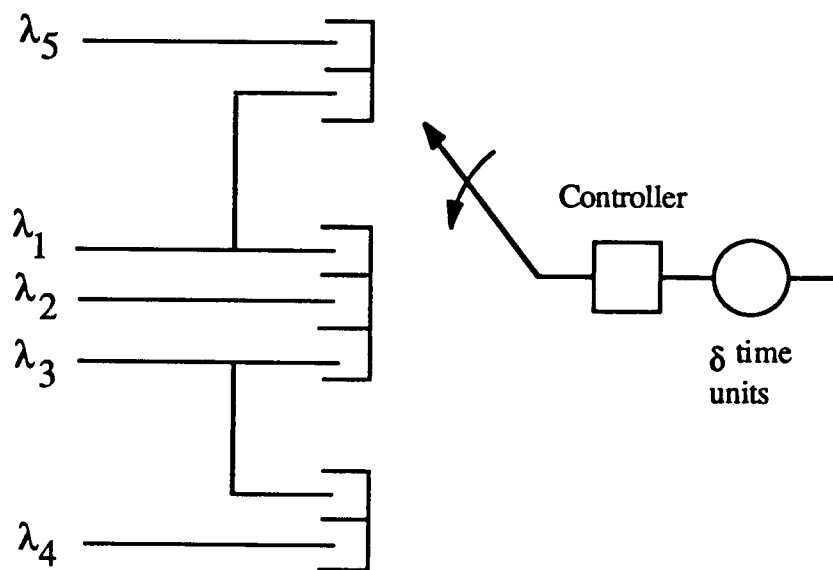


Figure 4b: 3 Class Polling Model

4), when establishing a circuit, we assume that a call needs to reserve a transceiver at the source and intermediate nodes only, and not at the destination node, which does not have to transmit¹¹. Therefore, in Figure 4, multihop calls simultaneously reserve the resource at two nodes (e.g., for a c_4 call we reserve a transceiver at nodes 1 and 2), whereas all other calls only reserve transceivers at one (source) node.

Call admission is concerned with the decision to accept or reject a call. We consider a simple threshold based call admission policy where a ' c_i voice call' is accepted if and only if there are less than t_i calls currently active over circuit i . In addition we require the policy to be work conserving, i.e., a c_i call is always assigned a transceiver if the threshold is not exceeded. Finally the thresholds t_i are chosen so as not to violate the physical constraints (limited number of transceivers per node) of the system. The latter can be expressed as linear constraints on the thresholds. The optimization problem is then to select the values of t_i so as to minimize the system performance subject to the system constraints

$$\begin{aligned} t_1 + t_4 &= n \\ t_2 + t_4 + t_5 &= n \\ t_3 + t_5 &= n \end{aligned}$$

To solve for the optimal thresholds, we convert the tandem queueing model to a multiclass polling model. Let the transceivers be sequentially indexed, and observe that c_1 and c_5 calls can be assigned the same transceiver since their respective paths do not overlap. Therefore, by analogy with the polling model, c_1 and c_5 constitute a *transmission set*. Similarly we can define two more transmission sets as $T_2 = \{c_1, c_2, c_3\}$ and $T_3 = \{c_3, c_4\}$. The polling model is shown in Figure 4b. Our objective is to select the optimal frame $s^* = [n_1^*, n_2^*, n_3^*]$ so as to minimize the average network blocking, where n_i is the number of transceiver assigned to transmission set i .

Finally, observe that the threshold associated with any call type is given by the total number of transceivers assigned to that call type. That is,

$$t_i = \sum_{k: i \in T_k} n_k$$

Therefore, the optimal thresholds are given by the following:

$$\begin{aligned} t_2^* &= n_2^* \\ t_4^* &= n_3^* \\ t_5^* &= n_1^* \\ t_1^* &= n_1^* + n_2^* \\ t_3^* &= n_2^* + n_3^* \end{aligned}$$

We have performed 2 sets of single-run optimization experiments. In each case, the initial slot assignment vector is $\theta^{(0)} = [0.4, 0.2, 0.4]$ and the system is observed for 50 parameter updates. We employ a gradient based optimization algorithm with a constant step size $\eta_k = 0.25$ and increasing

¹¹We are in effect assuming that receivers are readily available, and that only transmitters are a scarce resource.

observation interval. In particular $\tau_k = \tau_{k-1} + C$ with $C = 200$ calls. The system parameters for the experiments are given as:

Parameter Set 1: 24 Transceivers per Node

- Poisson Arrivals: $\lambda_1 = \lambda_3 = 0.1; \lambda_2 = \lambda_4 = \lambda_5 = 0.2$
- Call Durations: $\eta_4 = \eta_5 = 5; \eta_1 = \eta_2 = \eta_3 = UNI(1, 9)$
- Initial Observation Interval (τ_0): 2000 Call Arrivals

Parameter Set 2: 24 Transceivers per Node

- Poisson Arrivals: $\lambda_i = 0.2 \quad \forall i = 1, \dots, 5$
- Call Durations: $\eta_4 = \eta_5 = 5; \eta_1 = \eta_2 = \eta_3 = UNI(1, 9)$
- Initial Observation Interval (τ_0): 2000 Call Arrivals

Define the load on circuit i to be $\rho_i \equiv \lambda_i / \bar{\eta}_i$ where $\bar{\eta}_i = 1 / \eta_i$. In experiments 1 and 2 we assume 24 transceivers per node (equivalently T1 links between nodes) and consider both asymmetric and symmetric traffic in the network respectively. Moreover, for both parameter sets we assume that c_4 and c_5 calls have constant duration (5 frames), whereas all remaining call durations are uniformly (recall that we permit only integral frame lengths) distributed with mean 5.

Tables 2 - 6 give the system blocking, frame design and the corresponding slot assignments at the n th iteration, for the aforementioned parameter sets. For each parameter set we consider several initial frame configurations corresponding to different \mathbf{r} vectors. From the results we can make several observations; first for blocking networks with symmetric traffic (i.e., ρ_i is the same for each circuit), the optimal policy is to never accept a c_4 or c_5 call (see Table 5 - 6)¹². The corresponding optimal thresholds are therefore $t_1 = t_2 = t_3 = n; t_4 = t_5 = 0$. This is to be expected since both c_4 and c_5 calls reserve a transceiver at two nodes along the multihop path, whereas all other calls reserve only one transceiver. Therefore by accepting a c_4 call (and thereby reserving a transceiver at nodes 1 and 2) we are automatically blocking a c_1 and c_2 call, whereas by blocking a c_4 call we have the possibility of accepting two calls. Now since each call is of the same priority, and the arrival rates are the same, the optimal policy is never to accept c_4 calls. A similar argument hold for c_5 calls. However once the load is different on different circuits, the optimal policy is no longer to always reject multihop calls (see Table 1-3). Secondly, θ is optimized conditioned on an initial vector \mathbf{r} . Thus, we observe that although in Tables 1 and 3 the initial frame is common, θ^* is different for each case. However as is conjectured, the optimal frame is the same for each pair $(\theta_k^*, \mathbf{r}_k)$.

¹²We conjecture that this observation holds true irrespective of the actual value of the load. Therefore, even when ρ_i is sufficiently small such that the uncontrolled (i.e., subject to no admission control) network can sustain all offered traffic, the optimal policy is to never accept a c_4 or c_5 call.

11 Conclusions and Future Work

We have considered the *transmission scheduling problem* in an N -node Radio Network with homogeneous traffic (exclusively voice or data). The scheduling problem is modelled as a multiclass polling problem where a single resource must provide service to M predefined transmission sets. Customers arrive in one of several arrival streams, and a customer from a particular arrival stream is allowed to join one of several prespecified transmission sets associated with that stream. The traffic Grade-of-Service requirements lead to different scheduling policies, in particular, we consider a random polling policy for data traffic and a cyclic polling policy for voice traffic. We develop derivative estimation algorithms and use them in conjunction with a gradient-based optimization scheme to adaptively alter the schedule (in a stochastic operating environment) to achieve the desired system performance. An important feature of the proposed methodology is that it is independent of the nature of the node arrival processes (and arrival rate information, except for checking that no stability requirements are violated). Finally, for the case of data traffic, we use the optimal slot assignment probabilities to construct a *cyclic* or *deterministic* scheduling policy, using the Golden Ratio policy [15].

In the context of voice call scheduling, the proposed optimization scheme can be viewed as a new approach to solve a class of resource allocation problems with combinatorially hard stochastic features. Also, the underlying ideas for derivative estimation using marking and phantomizing techniques apply to general performance measures – not just the waiting times or blocking probabilities. Therefore, estimates of more general performance measures (e.g., probability that a customer delay exceeds some deadline, or probability that the queue length at a node exceeds some value) can also be obtained. This raises the interesting possibility of investigating scheduling problems involving traffic classes with different performance requirements, an issue of considerable importance in networks designed to support voice, video and data (see [24]).

Another natural next step for our work is to address the problem of routing in multihop radio networks. This can be done by defining appropriate transmission sets and scheduling packet transmissions accordingly. Although this problem becomes considerably more difficult when the network combines both voice and data, we believe the same basic approach described in this report can be applied.

Finally, it is obviously of great interest to investigate distributed implementations of the various algorithms we have presented for transmission scheduling. We believe this is indeed possible, but have left this research direction for future work.

Experiment 1: T1 Link with Asymmetric Traffic:

<i>Para. Set</i>	<i>r</i>	<i>n</i>	$s^{(n)}$	$\theta^{(n)}$	$P^{(n)}$
1	r_1	0	(14, 4, 6)	[0.400, 0.200, 0.400]	0.537
1	r_1	10	(7, 11, 6)	[0.202, 0.398, 0.400]	0.510
1	r_1	20	(6, 12, 6)	[0.187, 0.456, 0.357]	0.509
1	r_1	30	(6, 12, 6)	[0.153, 0.508, 0.339]	0.508
1	r_1	40	(5, 14, 5)	[0.127, 0.546, 0.327]	0.506
1	r_1	50	(5, 14, 5)	[0.124, 0.550, 0.326]	0.508

Table 2: Optimization of 4-node Tandem RN

<i>Para. Set</i>	<i>r</i>	<i>n</i>	$s^{(n)}$	$\theta^{(n)}$	$P^{(n)}$
1	r_2	0	(10, 5, 9)	[0.400, 0.200, 0.400]	0.513
1	r_2	10	(8, 9, 7)	[0.264, 0.421, 0.315]	0.512
1	r_2	20	(7, 11, 6)	[0.208, 0.542, 0.250]	0.511
1	r_2	30	(6, 12, 6)	[0.149, 0.612, 0.239]	0.507
1	r_2	40	(5, 14, 5)	[0.145, 0.631, 0.224]	0.507
1	r_2	50	(5, 14, 5)	[0.143, 0.634, 0.223]	0.506

Table 3: Optimization of 4-node Tandem RN

<i>Para. Set</i>	<i>r</i>	<i>n</i>	$s^{(n)}$	$\theta^{(n)}$	$P^{(n)}$
1	r_3	0	(14, 4, 6)	[0.400, 0.200, 0.400]	0.538
1	r_3	10	(6, 12, 6)	[0.174, 0.426, 0.400]	0.508
1	r_3	20	(6, 12, 6)	[0.160, 0.482, 0.358]	0.509
1	r_3	30	(5, 13, 6)	[0.133, 0.533, 0.334]	0.507
1	r_3	40	(5, 13, 6)	[0.135, 0.582, 0.283]	0.507
1	r_3	44	(5, 14, 5)	[0.137, 0.594, 0.269]	0.509
1	r_3	50	(5, 14, 5)	[0.137, 0.595, 0.268]	0.508

Table 4: Optimization of 4-node Tandem RN

Experiment 2: T1 Link with Symmetric Traffic:

<i>Para. Set</i>	\mathbf{r}	n	$\mathbf{s}^{(n)}$	$\theta^{(n)}$	$P^{(n)}$
2	\mathbf{r}_1	0	(14, 4, 6)	[0.400, 0.200, 0.400]	0.579
2	\mathbf{r}_1	5	(6, 13, 6)	[0.155, 0.445, 0.400]	0.516
2	\mathbf{r}_1	10	(4, 16, 4)	[0.080, 0.640, 0.280]	0.499
2	\mathbf{r}_1	15	(0, 21, 3)	[0.000, 0.784, 0.216]	0.479
2	\mathbf{r}_1	20	(0, 21, 3)	[0.000, 0.876, 0.124]	0.471
2	\mathbf{r}_1	25	(0, 21, 3)	[0.000, 0.935, 0.065]	0.479
2	\mathbf{r}_1	30	(0, 24, 0)	[0.000, 0.995, 0.005]	0.469

Table 5: Optimization of 4-node Tandem RN

<i>Para. Set</i>	\mathbf{r}	n	$\mathbf{s}^{(n)}$	$\theta^{(n)}$	$P^{(n)}$
2	\mathbf{r}_2	0	(10, 5, 9)	[0.400, 0.200, 0.400]	0.567
2	\mathbf{r}_2	5	(8, 9, 7)	[0.257, 0.442, 0.301]	0.541
2	\mathbf{r}_2	10	(5, 14, 5)	[0.131, 0.653, 0.216]	0.508
2	\mathbf{r}_2	15	(0, 20, 4)	[0.060, 0.824, 0.116]	0.483
2	\mathbf{r}_2	20	(0, 23, 1)	[0.060, 0.930, 0.010]	0.473
2	\mathbf{r}_2	21	(0, 24, 0)	[0.060, 0.640, 0.000]	0.471

Table 6: Optimization of 4-node Tandem RN

References

- [1] C.M. Barnhart, J.E. Wieselthier, and A. Ephremides, 'An Approach to Voice Admission Control in Multihop Wireless Networks,' *Proceedings of INFOCOM 1993*.
- [2] X.R. Cao, 'Convergence of Parameter in a Stochastic Environment,' *IEEE Transactions on Automatic Control* AC-30, 834-843, 1985.
- [3] C.G. Cassandras and V. Julka, 'Adaptive Algorithms For Scheduling Transmissions in Packet Radio Networks,' *Technical Report, University of Massachusetts*, 1992.
- [4] C.G. Cassandras and V. Julka, 'Marked/Phantom Algorithms for a Class of Scheduling Problems,' *Proc. of 31st IEEE CDC*, pp. 3215-3220, 1992.
- [5] C.G. Cassandras and V. Julka, 'Optimal Scheduling in Systems with Delay-Sensitive Traffic,' *to appear in Proc. of 32nd IEEE CDC*, 1993.
- [6] C.G. Cassandras and V. Julka, 'A New Approach For Some Combinatorially Hard Stochastic Optimization Problems Systems with Delay-Sensitive Traffic,' *Proceedings of the Allerton Conference*, 1993.
- [7] C.G. Cassandras and V. Julka, 'Scheduling Policies Using Marked/Phantom Algorithms,' *submitted for publication to Queueing Systems*.
- [8] E. K. P. Chong, and P. J. Ramadge, 'Convergence of Recursive Optimization Algorithms using IPA Derivative Estimates,' *Journal of Discrete Event Dynamics Systems*, Vol. 1, pp. 339-372, 1992.
- [9] I. Cidon, and M. Sidi, 'Distributed Assigned Algorithms for Multi-hop Packet-Radio Networks,' *IEEE Trans. on Computers*, vol. 38, 10, 1989.
- [10] A. Ephremides and T.V. Truong, 'Scheduling Broadcasts in Multihop Radio Networks,' *IEEE Trans. on Comm.*, vol. 38, no. 4, Apr. 1990.
- [11] P. Glasserman, '*Gradient Estimation via Perturbation Analysis*', Kluwer Academic Publishers, Boston, 1991.
- [12] P. Glynn, 'Likelihood Ratio Gradient Estimation An Overview,' *Proceedings of the 1987 Winter Simulation Conference*, pp. 336-375, 1987.
- [13] H. Heffes and D. Lucantoni, 'A Markov Modulated Characterization of Voice and Data and Related Statistical Multiplexer Performance,' *IEEE Journal on Selected Areas in Communications* SAC-4(6), pp. 856-867, August 1984.
- [14] Y.C. Ho, and X. Cao, 'Perturbation Analysis of Discrete Event Dynamic Systems,' Kluwer Academic Publishers, Boston, 1991.
- [15] Alon Itai and Zvi Rosberg, 'A Golden Ratio Policy for a Multiple-Access Channel,' *IEEE Transactions on Automatic Control* Vol. AC-29, No. 8, August 1984.

- [16] A.M. Law, and W.D. Kelton, 'Simulation Modeling and Analysis,' McGraw-Hill, New York, 1991.
- [17] B. Mohanty and C.G. Cassandras, 'The Effect of Model Uncertainty on Some Optimal Routing Problems,' *J. of Optimization Theory and Applications*, vol. 77, 2, pp. 256-290, 1993.
- [18] R. Nelson and L. Kleinrock, 'Spatial TDMA: A Collision-Free Multihop Channel Access Protocol,' *IEEE Trans. on Comm.*, COM-33, pp. 934-944, 1985.
- [19] M. Reiman and A. Weiss, 'Sensitivity Analysis for Simulations via Likelihood Ratios,' *Operations Research*, vol. 37, pp. 830-844, 1989.
- [20] Z. Rosberg, and M. Sidi, 'TDM Policies in Multistation Packet Radio Networks,' *IEEE Trans. on Comm.*, COM-37, 1, 1989.
- [21] P. Dupuis and R. Simha, 'On Sampling-controlled Stochastic Approximation,' *IEEE Transactions on Automatic Control*, Vol. AC-36, pp. 915-924, 1991.
- [22] H. Takagi, 'Queueing Analysis of Polling Models: An Update,' *Stoch. Anal. of Comp. and Comm. Syst.*, H. Takagi (Editor), 1990.
- [23] L. Tassiulas and A. Ephremides, 'Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks,' *Proc. of 29th IEEE Conf. on Decision and Control*, 1990.
- [24] J.E. Wieselthier, C.M. Barnhart and A. Ephremides, 'Optimal Admission Control in Circuit-Switched Multihop Radio Networks,' *Proc. of 31st IEEE CDC*, pp. 1011-1013, 1992.